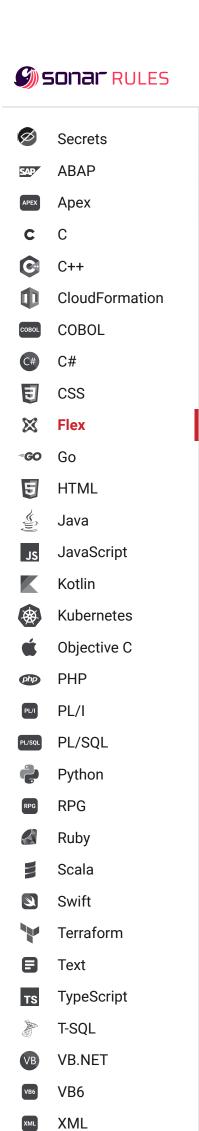
Q





Flex static code analysis

Unique rules to find Bugs, Security Hotspots, and Code Smells in your FLEX code

All rules 76	⊕ Bug
	Tags Search by name
Security.allowDomain() should only be used in a tightly focused manner Uulnerability	Track comments matching a regular expression Analyze your code
The flash.system.Security.exactSettings property should never be set to false Vulnerability	This rule template can be used to create rules which will be triggered when a comment matches a given regular expression. For example, one can create a rule with the regular expression .*REVIEW.*
Dynamic classes should not be used Code Smell	to match all comment containing "REVIEW". Note that, in order to match REVIEW regardless of the case, the (?i) modifier should be prepended to the expression, as in (?i).*REVIEW.*.
"LocalConnection" should be configured to narrowly specify the domains with which local connections to other Flex application are allowed	Available In: sonarqube
备 Vulnerability	
"default" clauses should be first or last Code Smell	© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved. Privacy Policy
Event types should be defined in metadata tags Code Smell	<u></u>
Event names should not be hardcoded in event listeners Code Smell	
The special "star" type should not be used	
Variables of the "Object" type should not be used	
Methods should not be empty	

Constant names should comply with a

All branches in a conditional structure should not have exactly the same

Classes that extend "Event" should

naming convention

Code Smell

implementation

Rug Bug

override "Event.clone()" • Bug
Constructors should not dispatch events
Rug
"ManagedEvents" tags should have companion "Event" tags
₩ Bug
Objects should not be instantiated inside a loop
Two branches in a conditional structure should not have exactly the same implementation
Constructor bodies should be as lightweight as possible
Only "while", "do" and "for" statements should be labelled
Statements, operators and keywords specific to ActionScript 2 should not be used
"for" loop stop conditions should be invariant
Unused function parameters should be removed
Code Smell